

Game State and Evaluator Report Testing and Work Plan

Testign Parser

Test for coverage of statement

has 100% method coverage and 96% line coverage except for line that throw exception for parser

Test case

```
t = t + 1 # keeping track of the turn number
m = 0 # number of random moves this turn
while (3 - m) { # made less than 3 random moves
  if (budget - 100) then {} else done # too poor to do anything else
  opponentLoc = opponent
  if (opponentLoc / 10 - 1)
  then # opponent afar
    if (opponentLoc % 10 - 5) then move upleft
    else if (opponentLoc % 10 - 4) then move downleft
    else if (opponentLoc % 10 - 3) then move down
    else if (opponentLoc % 10 - 2) then move downright
    else if (opponentLoc % 10 - 1) then move upright
    else move up
  else if (opponentLoc)
  then # opponent adjacent to this minion
    if (opponentLoc % 10 - 5) then {
      cost = 10 ^ (nearby upleft % 100 + 1)
      if (budget - cost) then shoot upleft cost else {}
    }
    else if (opponentLoc % 10 - 4) then {
      cost = 10 ^ (nearby downleft % 100 + 1)
      if (budget - cost) then shoot downleft cost else {}
    }
    else if (opponentLoc % 10 - 3) then {
      cost = 10 ^ (nearby down % 100 + 1)
      if (budget - cost) then shoot down cost else {}
    }
  }
```

```

}
else if (opponentLoc % 10 - 2) then {
    cost = 10 ^ (nearby downright % 100 + 1)
    if (budget - cost) then shoot downright cost else {}
}
else if (opponentLoc % 10 - 1) then {
    cost = 10 ^ (nearby upright % 100 + 1)
    if (budget - cost) then shoot upright cost else {}
}
else {
    cost = 10 ^ (nearby up % 100 + 1)
    if (budget - cost) then shoot up cost else {}
}
else { # no visible opponent; move in a random direction
    try = 0 # keep track of number of attempts
    while (3 - try) { # no more than 3 attempts
        success = 1
        dir = random % 6
        # (nearby <dir> % 10 + 1) ^ 2 is positive if adjacent cell in <dir> has no ally
        if ((dir - 4) * (nearby upleft % 10 + 1) ^ 2) then move upleft
        else if ((dir - 3) * (nearby downleft % 10 + 1) ^ 2) then move downleft
        else if ((dir - 2) * (nearby down % 10 + 1) ^ 2) then move down
        else if ((dir - 1) * (nearby downright % 10 + 1) ^ 2) then move downright
        else if (dir * (nearby upright % 10 + 1) ^ 2) then move upright
        else if ((nearby up % 10 + 1) ^ 2) then move up
        else success = 0
        if (success) then try = 3 else try = try + 1
    }
    m = m + 1
}
} # end while

```

```

oppoLoc = opponent
allyLoc = ally
if (oppoLoc) then
{
    if (oppoLoc / 10 - 1) then {
        if(oppoLoc % 10 - 5) then move upleft
    }
}

```

```

    else if(oppoLoc % 10 - 4) then move downleft
    else if(oppoLoc % 10 - 3) then move down
    else if(oppoLoc % 10 - 2) then move downright
    else if(oppoLoc % 10 - 1) then move upright
    else move up
  } else {
    if(oppoLoc % 10 - 5) then shoot upleft 1
    else if(oppoLoc % 10 - 4) then shoot downleft 1
    else if(oppoLoc % 10 - 3) then shoot down 1
    else if(oppoLoc % 10 - 2) then shoot downright 1
    else if(oppoLoc % 10 - 1) then shoot upright 1
    else shoot up 1
  }
}
else {
  move down
}

```

```

move up
move down
while (int) {if (int) then done else {}}
{{{if (T - t) then {T = T - t} else {T = T + t}}}}

```

Testign Evaluator and Gamestate

Only test for basic case of command with

```

oppoLoc = opponent
allyLoc = ally
if (oppoLoc) then
{
  if (oppoLoc / 10 - 1) then {
    if(oppoLoc % 10 - 5) then move upleft
    else if(oppoLoc % 10 - 4) then move downleft
    else if(oppoLoc % 10 - 3) then move down
    else if(oppoLoc % 10 - 2) then move downright
    else if(oppoLoc % 10 - 1) then move upright
  }
}

```

```

        else move up
    } else {
        if(oppoLoc % 10 - 5) then shoot upleft 1
        else if(oppoLoc % 10 - 4) then shoot downleft 1
        else if(oppoLoc % 10 - 3) then shoot down 1
        else if(oppoLoc % 10 - 2) then shoot downright 1
        else if(oppoLoc % 10 - 1) then shoot upright 1
        else shoot up 1
    }
}
else {
    move down
}

```

result in minions are able to move, attack and also able to call opponent, ally and nearby

and player could buy hex and spawn minion

Work Plan

Aa Name	 Date	 Contributor
<u>Refine Game State</u>	@February 6, 2025 → February 12, 2025	ไผ่
<u>Integrate Game State with Evaluator</u>	@February 13, 2025 → February 16, 2025	ไผ่
<u>Test Integrate</u>	@February 17, 2025 → February 18, 2025	ไผ่
<u>Design server-client</u>	@February 6, 2025 → February 16, 2025	บอณ
<u>Implement UI</u>	@February 6, 2025 → February 18, 2025	กัศ
<u>Test UI and Mock up</u>	@February 17, 2025 → February 18, 2025	บอณ